

WHAT IS CLAIMED IS:

- 1 1. A method of maintaining consistency of a plurality of data objects, each data object being
2 maintained by a different one of a plurality of computers that are operatively connected to
3 each other by a network, the method comprising, at a first one of the computers in
4 response to a received user input:
5 executing a current operation modifying a first one of the data objects that is maintained
6 by the first computer;
7 formatting a synchronization message, the message comprising:
8 a current operation parameter comprising data identifying said current operation,
9 data enabling another computer to execute said current operation in a manner
10 enabling synchronization of the first computer and the other computer, and
11 execution stage data identifying an operational stage for execution of the
12 current operation, and
13 a previous operation parameter comprising data identifying a preceding operation
14 executed by the first computer and execution stage data identifying an
15 operational stage at which said preceding operation was executed by the first
16 computer,
17 a parameter identifying a unique priority level of the first computer with respect
18 to others of the plurality of computers; and
19 transmitting the synchronization message from the first computer to others of the
20 plurality of computers.
- 21 2. The method of claim 1 wherein the previous operation parameter further comprises a
22 parameter identifying a one of the plurality of computers originating the preceding
23 operation, said computer being other than the first computer.

- 1 3. The method of claim 1 wherein transmitting from the first computer to others of the
2 plurality comprise transmitting using peer-to-peer network transmission.
- 3 4. The method of claim 3 wherein the method further comprises:
4 storing at the first computer a network identification of each of the other computers to
5 enable the transmitting by the first computer to each of the other computers.
- 6 5. The method of claim 5 wherein the parameter identifying a unique priority level
7 comprises a network identification of the first computer.
- 8 6. A data storage media comprising stored instructions for configuring a computer processor
9 to perform the method of claim 1.
- 10 7. A computer-implemented method of maintaining consistency of a plurality of data
11 objects, each data object being maintained by a different one of a plurality of computers
12 that are operatively connected to each other by a network, and the method being
13 performed at a first one of the plurality of computers to maintain a first one of the data
14 objects, the method comprising:
15 receiving a synchronization message from a second one of the plurality of computers, the
16 synchronization message comprising:
17 a current operation parameter comprising data identifying a first operation that
18 was executed by the second computer, data required for execution of the first
19 operation at the first computer to manipulate the first data object in a manner
20 effecting synchronization of the first and the second computer, a first
21 execution stage parameter identifying an operational stage at which the first
22 operation was executed by the second computer, and a value identifying the
23 second computer,

FOCUS at 9:25:07

1 a previous operation parameter comprising a parameter identifying a second
2 operation that was executed by the second computer prior to execution of the
3 first operation, a second execution stage parameter identifying an operational
4 stage at which said second operation was executed at the second computer,
5 and an identity of a computer originating said second operation,
6 a parameter identifying a unique priority level of the second computer with
7 respect to others of the plurality of computers; and
8 updating the first one of the data objects by executing the operation identified by the
9 current operation parameter at the first computer when the first computer is at a
10 correct execution stage.

- 11 8. The method of claim 7 wherein the identity of the computer originating said second
12 operation comprises a network identity of said originating computer.
- 13 9. The method of claim 7 wherein updating when the first computer is at a correct execution
14 stage further comprises:
15 comparing the current operation parameter and the previous operation parameters to local
16 state data stored at the first computer, the local state data comprising data indicating a
17 current execution stage and a last executed operation; and
18 executing the first operation when the local state data indicates that a current execution
19 stage at the first computer corresponds to the execution stage indicated by the first
20 execution stage parameter and the previous operation parameter identifies the last
21 operation executed at the first computer.
- 22 10. The method of claim 9 further comprising:
23 storing the synchronization message in a waiting list when the current execution stage at
24 the first computer is earlier than a execution stage for the operation identified by the
25 first execution stage parameter.

1 11. The method of claim 10 wherein storing the synchronization message further comprises:
2 deleting from the waiting list a synchronization message having a lower execution
3 priority and a same first execution stage parameter value as for the synchronization
4 message and a same second execution stage parameter value as for the
5 synchronization message.

6 12. The method of claim 9 wherein, when the local state data indicates that the first
7 computer is at an execution stage subsequent to the execution stage indicated by the first
8 execution stage parameter:
9 searching a journal of operations executed by the first computer to locate a journaled
10 operation executed at a same execution stage as indicated by the first execution stage
11 parameter, and
12 when said journaled operation has a lower execution priority than the execution priority
13 associated with the first synchronization message, undoing said journaled operation
14 and subsequent operations and then executing the operation identified by the first
15 synchronization message.

16 13. The method of claim 12 wherein undoing said journaled operation and subsequent
17 operations comprises undoing only when the previous operation parameter identifies a
18 same operation as was executed prior to execution of said journaled operation.

19 14. The method of claim 12 wherein undoing operations comprises storing each undone
20 operation in a waiting queue to enable subsequent re-execution of said undone
21 operations.

1 15. The method of claim 12 further comprising:

2 when said journaled operation has a higher execution priority than the execution priority
3 associated with the first synchronization message, storing the first synchronization
4 message in the waiting list.

5 16. A data storage media comprising stored instructions for configuring a computer processor
6 to perform the method of 7.

7 17. A computer system comprising:

8 a processor;
9 a network interface coupled to the processor and operatively coupling the computer
10 system to a plurality of other computer systems;
11 a memory coupled to the processor and storing a first data object, said first data object
12 being a local replica of data replicated at others of the plurality of computers;
13 a local state module configured to record data effecting manipulations of the first data
14 object, said local state module comprising memory recording operations effecting
15 synchronization of the first data object; and
16 a consistency module coupled to the processor and memory and operating to maintain
17 consistency of the first data object with replicas of the data object that are stored at
18 the other computer systems, said maintaining being performed based on processing of
19 received synchronization messages, each received synchronization message
20 comprising:
21 a current operation parameter comprising data identifying a first operation that
22 was executed by another computer, data required for execution of the first
23 operation so as to manipulate the first data object in a manner effecting
24 synchronization of the first data object with a data object stored at the other
25 computer, a first execution stage parameter identifying an operational stage at
26 which the first operation was executed by the other computer, and a value
27 identifying the other computer, and

1 a previous operation parameter comprising a parameter identifying a second
2 operation that was executed by the other computer prior to execution of the
3 first operation, a second execution stage parameter identifying an operational
4 stage at which said second operation was executed at the other computer, and
5 an identity of a computer originating said second operation,

6 18. The system of claim 17 wherein the local state module comprises stored instructions to
7 configure the processor to:
8 compare the current operation parameter and the previous operation parameter to data
9 maintained by the local state module indicating a current execution stage of the
10 system with respect to manipulations of the first data object and a last operation
11 effecting the first data object;
12 instruct the consistency module to update the first data objects by performing an
13 operation identified by the current operation parameter when the system is at a correct
14 execution stage;
15 store the synchronization message in a waiting list when the system is at an earlier
16 execution stage than that identified by the first execution stage parameter; and
17 delete from the waiting list a synchronization message having a lower execution priority
18 and a same first execution stage parameter value as for the synchronization message
19 and a same second execution stage parameter value as for the synchronization
20 message.